. **Amendments to the Claims:**

1-38. (Canceled)

39. (New) A generic device controller system, comprising:

a processor, comprising standard non-true real time peripheral device control software;

at least one peripheral device requiring true real time peripheral device control; and

a generic device controller, coupled between the processor and the at least one peripheral device, for providing true real time peripheral device control for the at least one peripheral device in response to non-true real time control requests from the processor.

40. (New) The system of Claim 39, wherein:

the processor does not provide true real time peripheral device control; and

the generic device controller provides true real time peripheral device control.

41. (New) The system of Claim 40, wherein:

the processor does not comprise a true real time kernel; and

the generic device controller provides true real time peripheral device control.

42. (New) The system of Claim 40, wherein:

the processor does not comprise a layered true real time operating system; and

the generic device controller produces true real time peripheral device control.

43. (New) The system of claim 39 wherein:

the processor and the generic device controller communicate using a single universal communications stream;

the at least one peripheral device communicates using a data protocol and associated data specific to the at least one peripheral device; and

the generic device controller:

converts between the single universal communications stream and the specific protocol and associated data for communicating with the at least one peripheral device, and

communicates with the at least one peripheral using the specific data protocol and associated data.

44. (New) The system of claim 43 wherein the specific data protocol and associated data is one of: (a) I^2C, (b) RS-232 serial port, (c) RS-422/RS-485 serial port, (d) LPT parallel printer port, (e) 8-bit bi-directional data ports, and (f) general purpose I/O port interface.

45. (New) The system of claim 43, wherein the universal communications stream is a multi-drop communications protocol.

46. (New) The system of claim 45, wherein the universal communications stream is one of: (a) ATM, (b) Ethernet, (c) CAN, (d) I2C, and (e) multi-drop serial communications.

47. (New) The system of claim 41, wherein the universal communications stream is a network protocol.

48. (New) The system of claim 47 wherein the universal communications stream is one of: (a) Ethernet, (b) ATM, (c) WAN, (d) Infrared, (e) Serial, and (f) fiber optics.

49. (New) The system of claim 41, wherein the universal communications stream comprises a Universal Serial Bus (USB) data and communications protocol.

50. (New) The system of claim 49 further comprising a USB hub coupled between the processor and the generic device controller.

51. (New) The system of claim 39, wherein the processor comprises a non-true real time operating system.

52. (New) The system of claim 51 wherein the non-true real time operating system is one of: (a) Windows NT, (b) Windows 98, (c) Windows 2000, (d) LINUX, (e) WinCE, (f) QNX, (g) DOW, (h) VXWorks, (i) Whistler, and (j) Whistler embedded.

53. (New) The system of claim 51 wherein the non-true real time operating system is a Win32 environment.

54. (New) The system of claim 53 wherein the non-true real time operating system if one of: (a) Windows NT, (b) Windows 98, (c) Windows 2000, and (d) WinCE.

55. (New) The system of claim 39 implemented in at least one of a plurality of networked gaming machines.

56. (New) In a generic device controller system, comprising a processor, at least one peripheral device, and a generic device controller coupled between the processor and the at least one peripheral device, a method comprising:

running non-true real time peripheral device control software in the processor;

requiring true real time peripheral device control in the at least one peripheral device; and

providing true real time peripheral device control in the generic device controller for the at least one peripheral device in response to non-true real time control from the processor.

57. (New) The method of claim 56 wherein:

the processor does not provide true real time peripheral device control; and

the generic device controller provides true real time peripheral device control.

58. (New) The method of claim 56 further comprising:

communicating between the processor and the generic device controller using a single universal communications stream;

communicating with the at least one peripheral device using a data protocol and associated data specific to the at least one peripheral device;

converting in the generic device controller between the single universal communications stream and the protocol and associated data specific to the at least one peripheral device; and

communicating between the generic device controller and the at least one peripheral device using the data protocol and associated data specific to the at least one peripheral device.

59. (New) The method of claim 58 wherein the converting step comprises converting in the generic device controller between the single universal communications stream and one of: (a) I^2C, (b) RS-232 serial port, (c) RS-422/RS-485 serial port, (d) LPT parallel printer port, (e) 8-bit bi-directional data ports, and (f) general purpose I/O port interface.

60. (New) The method of claim 58 wherein the step of communicating between the processor and the generic device controller comprises using a multi-drop communications protocol.

61. (New) The method of claim 60 wherein the multi-drop communications protocol comprises using one of: (a) ATM, (b) Ethernet, (c) CAN, (d) I2C, and (e) multi-drop serial communications.

62. (New) The method of claim 58 wherein the step of communicating between the processor and the generic device controller comprises using a network protocol;

63. (New) The method of claim 62 wherein the step of using a network protocol comprises using one of: (a) Ethernet, (b) ATM, (c) WAN, (d) Infrared, (e) Serial, and (f) fiber optics.

64. (New) The method of claim 58 wherein the step of communicating between the processor and the generic device controller comprises using a Universal Serial Bus (USB) data and communications protocol.

65. (New) The method of claim 64 wherein the step of communicating between the processor and the generic device controller further comprises coupling the processor and the generic device controller through a USB hub.

66. (New) The method of claim 56 wherein the step of running non-true real time peripheral device control software in the processor comprises running a non-true real time operating system in the processor.

67. (New) The method of claim 66 wherein the step of running a non-true real time operating system in the processor comprises running one of: (a) Windows NT, (b) Windows 98, (c) Windows 2000, (d) LINUX, (e) WinCE, (f) QNX, (g) DOW, (h) VXWorks, (i) Whistler, and (j) Whistler embedded.

68. (New) The method of claim 66 wherein the step of running a non-true real time operating system in the processor comprises running a non-true real time operating system in a Win32 environment.

69. (New) The method of claim 68 wherein the step of running a non-true real time operating system in a Win32 environment comprises running one of: (a) Windows NT, (b) Windows 98, (c) Windows 2000, and (d) WinCE.

70. (New) The method of claim 56 implemented in at least one of a plurality of networked gaming machines.